

# 10.4 Sydney - Release 1

Updated RAD Studio 10.4.1 Sydney release available (September 2<sup>nd</sup>, 2020).



**Note:**

- To see what is included in **10.4 Sydney**, [click here](#).

RAD Studio 10.4 Sydney - Release 1 (also known as 10.4.1) is available for installation. 10.4.1 builds on the feature set of 10.4, enhancing existing features throughout the product, and adding new capabilities. Delphi 10.4.1, C++Builder 10.4.1, and RAD Studio 10.4.1 are available to download to any active Update Subscription customer.

RAD Studio 10.4.1 has a strong focus on quality improvements. Key quality focus areas include:

- IDE
- Delphi Code Insight (LSP)
- Parallel Library
- SOAP & XML
- C++ Toolchain
- FireMonkey
- VCL
- Delphi Compiler
- iOS Deployment

RAD Studio 10.4.1 includes all fixes from 10.4 Patch 1 ([http://docwiki.embarcadero.com/PlatformStatus/en/Main\\_Page#RAD\\_Studio\\_10.4\\_Patch\\_1](http://docwiki.embarcadero.com/PlatformStatus/en/Main_Page#RAD_Studio_10.4_Patch_1)), Patch 2 ([http://docwiki.embarcadero.com/PlatformStatus/en/Main\\_Page#RAD\\_Studio\\_10.4\\_Patch\\_2](http://docwiki.embarcadero.com/PlatformStatus/en/Main_Page#RAD_Studio_10.4_Patch_2)), and Patch 3 ([http://docwiki.embarcadero.com/PlatformStatus/en/Main\\_Page#RAD\\_Studio\\_10.4\\_Patch\\_3](http://docwiki.embarcadero.com/PlatformStatus/en/Main_Page#RAD_Studio_10.4_Patch_3))

In addition to quality enhancements, 10.4.1 also includes a number of new features, as detailed below.

## Installing 10.4.1

If you have already installed 10.4 Sydney (May 26th, 2020), installing 10.4 Sydney - Release 1 requires a full uninstall and reinstall. As part of the uninstall process, you will see an option for preserving your settings. This is enabled by default. Separately, you can also preserve settings with [Settings Migration Tool](#).

## Key Enhancements by product area in 10.4.1

## Supported Platforms

RAD Studio now supports deploying apps on Windows Server 2019.

## Contents

- Installing 10.4.1**
- Key Enhancements by product area in 10.4.1**
  - Supported Platforms
  - IDE
    - Projects
    - Layouts
    - UI
    - General
    - Smaller but notable fixes
  - Delphi LSP/ Code Completion Changes
    - Code Insight for Large Projects
  - RTL Improvements
    - New TLightweightMREW Record
    - Parallel Programming Library

# IDE

## Projects

- Package AUTO libsuffix: packages can now have an automatic version suffix, instead of manually updating and specifying the right version suffix with each new release. (The compiler quietly supported this in 10.4, but full support for the feature in the IDE and package project settings is introduced in 10.4.1.)
- The Pentium-safe FDIV compiler option is no longer shown in the Delphi compiler options.
- The Projects pane context menu items that were no longer available after 10.3 (invoked by right-clicking on an empty area in the Projects pane) are now available via a drop-down in the Projects pane toolbar.

## Layouts

- Improved support for multi-monitor desktop layouts for those using multiple editor windows across multiple screens
- The floating form designer has been removed. Note that you can still have multiple edit windows including a form designer in a second window, and in fact, support for that has been enhanced.
- When the IDE switches layout, such as to/from debugging, or to/from startup (e.g. when opening or closing a project), if the same layout is specified for both states, the layout won't be applied. This avoids unnecessarily switching layouts.
- Desktop layouts which cannot be applied (i.e. are floating layouts, no longer supported, see above) will be listed in gray in the Desktop Layout combo box in the title bar. When a floating layout is applied, the IDE will instead show a dialog.
- Improved support for 4K monitors: IDE now correctly restores layouts on a 4K monitor in situations where the 4K screen is not scaled
- If the IDE expects a project's .dsk file but it does not find it, it opens using the default layout.

## UI

- Options and other dialogs:
  - The Project Options, IDE Options, GetIt, and New Items dialogs now have scrollbar tracking (will move as you drag the scrollbar.) This includes scrolling with the mouse wheel
  - The Options dialog (environment options) used to always open to show settings for the Win64 target platform; now it opens according to the currently active platform.
- Several listview controls, which can be difficult to read in the dark theme, have been replaced by the virtual treeview.
- The guided tour was removed from the Welcome page
- Several Object Inspector fixes, especially around flickering and selecting the wrong property
- Improved the selection of items on the Structure pane, including where you would click on an item and another item was selected

## General

- 'Broadcast to devices' is off by default. This always accessed the network on IDE startup, causing errors for some customers. Now that it's disabled, (valid) network errors will only occur if you turn it on.
- The names of the 64-bit Windows debuggers (visible in the Attach To Process dialog) have been clarified to explain which is for Delphi and which for C++. This should help you choose which debugger to use when attaching to a process.

## Smaller but notable fixes

- 'Delete Invalid Paths' should no longer remove valid paths.
- Moving toolbars no longer presents issues.
- "-showotamenunames" (used by ToolsAPI developers) has been re-enabled.
- Editing a string list in the code editor now works again.
- Dialogs are shown in the IDE using a modern style.
- Drawing issues, such as showing the IDE window as a solid color when opening a project, are resolved.
- On Windows 7, the title bar no longer has a white line at the top.
- The action editor used to draw images as black, which is resolved.

SOAP and XML

Web Clients

Database

Other RTL Improvements

VCL Improvements

FireMonkey

FMX TMem improvements

Smaller but notable fixes

Delphi Language

Managed Records

C++ Language

Win64 Debugger

Linker

Exception Handling

General Quality

Other Enhancements

See Also

- Some IDE menus now draw correctly (e.g. popup menus in Audits); also some menus did not show the menu Checked state, but now do.
- The debug symbols search path now supports \$(Platform) and \$(Config) environment variables
- The code editor used to let you select invalid encodings for a unit (applicable only to forms); these are now no longer available.
- An issue installing packages that presented as being unable to load a package, caused by packages with the filename case differing on disk and usually seen with large projects, has been resolved.

## Delphi LSP/ Code Completion Changes

- Code Insight now supports completing in a class declaration
- Significant performance work:
  - Improvement in memory usage, with DelphiLSP processes using much less memory
  - Improvement in speed: code completion results should appear much more quickly
- Inside Code completion, go to definition and hover, all provide better results (e.g., code completion improves the list of symbols it suggests.)
- There are new options for controlling code completion, allowing you to tweak the functionality to behave much more similarly to classic completion. Read more about these settings [here](#).
- When filtering code completion as you type, the matching characters are now optionally drawn underlined.
- Class completion settings have been moved to the Code Insight settings page
- Files changed outside the IDE (e.g. by an external editor or source control) will be recognized by the LSP server and re-scanned.
- Several fixes for general Code Insight issues presenting as no completion results, including an issue with a + symbol in the path, plus several others.
- Improved handling of packages
  - Packages with contained units no longer cause errors
  - Code completion will offer symbols from a required package
  - There should be no more false errors reported in packages.
  - Code completion in packages, in general, is improved.
- Help Insight shows accurate content when hovering over a symbol.
- Code completion will work when 'Use MSBuild externally to compile' is turned on.c
- Errors rendering 'red squiggles' have been resolved when the length of the red squiggly line was wrong.
- When code completion's list was set to sort by name instead of scope, selecting an item inserted the wrong text. This is fixed.
- 10.4.1 also includes all fixes from 10.4 Patch 2, such as not inserting () braces when they already exist, pressing Space not selecting the current completion, completion lagging, and then not entering text correctly, Error Insight in the Structure pane lagging by a keystroke, and so forth.

## Code Insight for Large Projects

The following fixes will be especially noticeable for large projects. Note the above memory usage and performance fixes are also very significant for large projects.

- The compiler improved attempts to parse past fatal errors.
- Improved the notifications to the server when there are IDE-side changes. This reduces churn and improves completion accuracy.

## RTL Improvements

### New TLightweightMREW Record

We are introducing a new reader-writer lock implementation, working across all supported platforms and which is faster and more lightweight than the old TMultiReadExclusiveWriteSynchronizer (a.k.a. TMREWSync). TLightweightMREW wraps a native implementation of a reader-writer lock on each platform.

For more information, please read the TLightweightMREW documentation.

## Parallel Programming Library

Many fixes to TThread.WaitFor, TParallel.For, TParallel.Join, TConditionVariableMutex.WaitFor for Linux, TTask.WaitForAny, and the TPoolThread class IdleWorkerThreadCount. We also addressed some issues with TThreadedQueue related with PopItem, the use of a large number of threads, and more.

## SOAP and XML

- XML quality work: a few long-standing issues with the TXMLDocument component were improved, tied to the various DOMVendors we support (OmniXML, ADOM). Some of these issues are related to UTF support and encodings.
- SOAP WSDL Importer: improved the WSDL importing logic and also SOAP clients invocation.

## Web Clients

- Winapi.Winsock2 now includes inet\_ntop to better support IPv6 addresses
- TRESTResponseDataSetAdapter imports JSON strings with appropriate length
- Improved support for enums in TJSONSerializer
- A few TRestResponseDataSetAdapter fixes
- Creating a TIPAddress from a URL now works also on Windows
- The NSURLConnection API was deprecated by Apple on Mac OS X 10.11 and iOS 9, but it was still used in our HTTP Client Library. 10.4.1 introduces support for the newer NSURLSession API on both platforms. This change also addresses issues with proxy support on macOS.

## Database

- A significant number of DataSnap issues have been addressed, including some performance problems and memory leaks, MidasLib issues, DatasetProvider and TClientDataSet ones
- Improved FireDAC support for SQL Server Function Parameters
- Fix in TFDMemTable FindFirst with IndexFieldNames
- Improved TFDQuery with Calculated- and BLOB-Fields
- Better TFDTable PostgreSQL multicolumn index field order
- FireDAC-PostgreSQL parsing of QuoteTag inside strings

## Other RTL Improvements

- Performance improvements for TStrings with custom LineBreak
- Optimized TObject methods returning or checking strings
- Exposed UTF8IdentXxx functions
- Optimized LoadResString

## VCL Improvements

- Several TGrid regressions in 10.4 (partially already addressed in 10.4 Patch 2) have been resolved, including TStringGrid and other grid controls, misalignment in grid StretchDraw method in OnDrawCell, and others
- Improved TGridPanel's handling of percentage-based spacing. You can now specify exact percentages for columns or rows and have those apply instead of them being modified as you set them. This greatly improves using the TGridPanel layout control to get exactly the spacing your layout needs
- TBitBtn's inbuilt glyphs have been moved to HighDPI, plus scaling improvements
- The custom title bar supports high DPI scaling and has a number of other quality fixes
- TUpDown component Position property and OnChangingEx event improvements
- StatusBar component supports DoubleBuffered drawing
- TPageControl SetTabIndex methods now works correctly if there are hidden tabs
- TDateTimePicker text doesn't look blurred on move hovering
- TLabel word wrap has been improved by replacing the font height calculation logic
- TListView infotip stays visible longer
- DB Controls border looks the same of the non-data-aware controls
- Several style improvements
- Issues with the caption widths of owner drawn top-level menu items, and styled menus and their scaling, have been improved
- Styles respect the ImageMargins of TButton and TButton text positions

# FireMonkey

## FMX TMemo improvements

We are continuing the work on the new FireMonkey TMemo component architecture, by adding additional improvements to the Windows implementation and adding support for macOS to this new FMX TMemo component architecture. Most of the focus is on TMemo IME mode for Korean, Japanese, and Chinese (among other languages) but there are also general improvements in performance, stability, scrolling behavior, mouse selection, copy and paste, and more. There are also fixes for TEdit.

## Smaller but notable fixes

- The Storyboard Launch Screen has added support for icon sizes required by Apple for iOS.
- Improved permissions management for iOS Push notifications
- TListView OnItemClick fires even if already selected
- Fixed an issue with inconsistent values for TMediaPlayer.Duration on Android
- iOS WebBrowser issue with external links
- TStringGrid OnCellDblClick event
- TListView Checkmark Accessory disappearing

## Delphi Language

The 10.4.1 release improves the Delphi compiler in several areas. In particular the focus was on supporting the LSP code completion, and in improving the managed records quality, but we also addressed a few 10.4 regressions and unexpected changes (including issues with goto statements and other uncommon scenarios). There are some improvements in package references and C++ HPP files generation. The release also improves compiler performance in a few areas, optimizing unit names lookup and other internal searches, benefiting large projects. There are also a few code generation improvements (that is, cases in which the compiler can generate code that performs better).

## Managed Records

This release improves Managed Records quality, including:

- Return values optimizations
- Cleanup of a few compiler errors
- Improvements to operator Assign
- Changes to Default(T) for managed records
- Better coexistence of managed records and generics
- Some cases of missing invocation of Initialize

## C++ Language

## Win64 Debugger

- The Win64 LLDB-based debugger has several improvements, especially around catching and continuing from exceptions, and in performance (especially when there are many threads.)
- Fixed an issue where Delphi visualizers were being invoked: visualizers for TStringList, Currency, TDateTime are no longer invoked.
- Fixed an issue where the exception dialog was shown when using F7 to start a new debug session.
- Handling of OS exceptions is improved.
- IDE performance improved when stepping when there are hundreds of threads.

## Linker

- The Win64 linker (ilink64) has had significant revision of its memory handling. This should reduce out of memory errors in large projects, or with debug builds.

## Exception Handling

- Several key exception handling issues have been fixed. This includes an issue affecting ACE/TAO allocating memory, where the Windows API can throw a SEH exception.

## General Quality

- Several crashes, ICEs and similar resolved in the Clang compiler.
- Issues generating RTTI for closures have been resolved.
- Improvements in debug info generation especially for Delphi-style classes.
- Improvements in the RTL, mainly around removing the MFC compatibility macro and always exporting these symbols, as well as compatibility issues found while supporting common C++ libraries.

## Other Enhancements

- Addressed an access violation in the License Manager application.
- Directly invoking GetItCmd.exe no longer removes installed packages from the system

## See Also

- [Installation Notes](#)
- [Release Notes](#)
- [New features and fixed issues](#)
- [What Was New in Past Releases](#)

---

Retrieved from "[http://docwiki.embarcadero.com/RADStudio/Sydney/e/index.php?title=10.4\\_Sydney\\_-\\_Release\\_1&oldid=272053](http://docwiki.embarcadero.com/RADStudio/Sydney/e/index.php?title=10.4_Sydney_-_Release_1&oldid=272053)"

---

**This page was last edited on 2 September 2020, at 08:30.**